



برنامج النحميل Aria2

أكثر من خيار لتنزيل الملفات
إلى جهازك بقوة وداعم للتورنت

تعرف على لغة البرمجة بايثون
من أشهر لغات البرمجة التفسيرية

في نظام التشغيل جنو/لينكس

سَطْرُ الْأَوَامِرِ الْأَجْمَلِ
وَالْأَكْثَرِ مَتْعَةً وَفَائِدَةً



داخل العدد

1

أخبار سريعة

3

إفتتاحية العدد

4

برامج

5

تلميحات وأفكار

6

البرمجة

8

في طريق الإحتراف

9

فاكهة لينكس

11

شبكات

14

الأمن في لينكس

16

عالم الألعاب

مجلة لينكس العرب

مجلة شهرية شاملة تهتم بأمور لينكس خاصة والمصادر المفتوحة بشكل عام

العدد الثاني
ديسمبر ٢٠٠٧

”

أعد موضوعات هذا العدد
أعضاء موقع لينكس العرب
www.linux-ar.org

التصميم والإخراج الفني
أحمد السويلم
عبدالله الدعجاني

للمساهمة في الإعداد المقبلة
www.linux-ar.org/forum

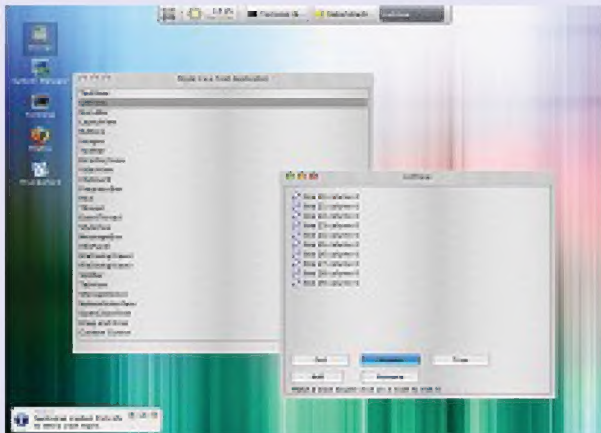
أو المراسلة على العنوان
linux1010@gmail.com

مستخدمي لينكس ويونكس، وقد تضمن الإعلان أن هذه الإصدارية يمكن استخدامها بعد أن تم حل الكثير من المشاكل والشواهد.

يذكر أن الإصدارات المرشحة تكون عادة شبه نهائية ويمكن الاعتماد عليها، حيث تسبقها مرحلتها ألفا وبيتا وهما فترات اختبار لأي برنامج أو نظام جديد.

الخط الزمني لتسلسل الإصدارات بين أن موعد إطلاق النسخة النهائية هو شهر ديسمبر من العام الحالي 2007.

SkyOS



■ صدرت مؤخراً نسخة تطويرية جديدة تحمل الرقم 6814 من نظام SkyOS. من أبرز المزايا الجديدة التي أدخلت عليها إضافة مدقق إملاتي يعمل على مستوى النظام بالكامل وكذلك تضمينها محرك موزيلا Gecko وإجراء تحديث لمدير الملفات وإضافة محرك جديد للسمات (المظهر) وتحديثات أخرى لا يتسع المجال لذكرها.

الجدير بالذكر أن SkyOS نظام تشغيلي كتب من الصفر وظهر لأول مرة عام 1996 حيث بدأ كنظام حر مجاني ومفتوح المصدر لكنه قد تحول فيما بعد إلى نظام تجاري مغلق. ولا يزال هذا النظام متوافقاً بشكل شبه كلي مع مواصفات POSIX، وقد بني على نواة وأدوات GNU لينكس ويونكس. لمزيد من التفاصيل زيارة الموقع التالي: <http://www.skyos.org>

شركة Sun تدعم برنامج بليندر



■ قررت شركة Sun Microsystems الشهيرة صاحبة نظام Solaris أن تدعم مشروع فيلم "خوخ" peach، وهو الفيلم الذي ينتجه تون روسيندل المطور الأول لبرنامج التصميم ثلاثي الأبعاد (بليندر).



تحديث ثان لجنوم 2.20

■ التحديث الثاني وقبل الأخير لسطح المكتب جنوم 2.20 تضمن جملة من التغييرات من أهمها إجراء تحديثات لهذه الإصدارية وإصلاح حالات الانهيار والأخطاء والشواهد البرمجية التي كانت تواجه المستخدمين، إضافة إلى إدخال بعض التعديلات على Eye of Gnome و Metacity و Glib و Gstreamer و tomboy و Epiphany و 2.20.1.

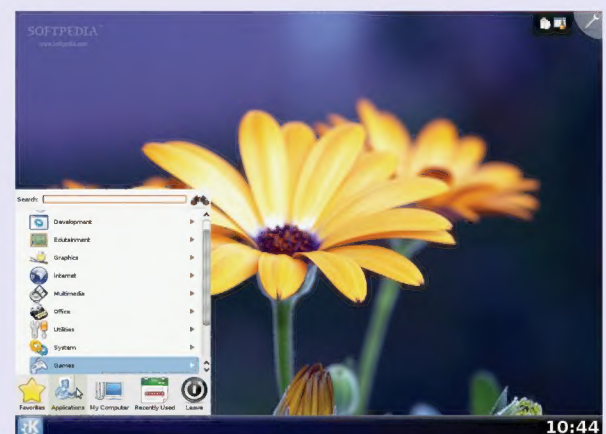
حزمه Koffice اصداره الفا 5

■ أعلن مؤخراً عن إطلاق النسخة ألفا خمسة من حزمه برامج المكتب كي أوفيس في إصدارته التجريبية الثانية. وتجدر الإشارة إلى أن هذه الحزمة إحدى أشهر تطبيقات المكتب التي تصدر عن مجتمع كي دي إي. كما تم إطلاق إصداره خاصه وبمبسطه للأطفال مع واجهه خاصه لتسهيل التعامل مع البرامج المرافقة لهذه الحزمة. إضافة إلى إدخال مميزات جديدة لبرنامج الرسم Krita.

للمزيد من المعلومات :

koffice.org/announcements/announce-2.0alpha5.php

KDE 4 RC1



■ اصدر مجتمع KDE الإصدارية المرشحة الأولى من واجهه كي دي إي والتي تحظى بشعبية كبيرة بين أوساط

أو التصوير النهائي للفيلم . هذه المرة تقدمت شركة Sun Microsystems بدعمها لإستخدام أجهزتها القوية جدا من فئة 64 بت و 4 جيجا بايت من الرامات و أجهزة ذات ثمانية معالجات و أجهزة شركة Sun Microsystems المبنية على معالجات SPARC وتستخدم نظام التشغيل Solaris. يقول تون روسيندل : سنستخدم نظام Solaris على جهاز واحد على الأقل فى الأستوديو لعمل التجارب و التجميع " Compile " و باقى الأجهزة يكون عليها نظام التشغيل لينكس. والمطورين يؤيدون نظام التشغيل لينكس و يقولون أنه أكثر دعما للأجهزة و البرامج .

تجدر الإشارة إلى أن هذه ليست التجربة الأولى، حيث سبق تنفيذ مشروع مماثل وهو أحلام الفيلة. الهدف من هذه المشروعات هو تطوير برنامج بليندر فخلال عملية صناعة الفيلم يكون معهم مطورى البرنامج و خاصة تون روسيندل و فى فيلم أحلام الفيلة إكتسب البليندر الكثير من المزايا التى ظهرت مع النسخة 2.42 . و فيلم خووخ الآن يتم عمله لتطوير النسخة القادمة 2.5 و التى ينتظر بها الكثير من المزايا التى ستجعل من البليندر برنامجا عملاقا ينافس أقوى البرامج التجارية بشدة . الفيلم السابق تم استخدام أجهزة من شركة apple لعمل الرندر

يكتبها - أحمد عبدالله السويلم

وفي الجانب الآخر نجد أن كثير من أسرى الأنظمة المغلقة مازالوا قابعين تحت سيطرتها ويتملكهم الخوف من ذلك الشعاع القوي الذي دخل إلى حياتنا التقنية، وهو شي منطقي، فالخوف والتردد قد تولد بسبب التشبع بثقافة المستخدم الذي سلبت حريته وخياراته، ويتعامل مع ما هو موجود بقله حيلة كواقع مفروض ليس من حقه الاعتراض أو محاولة التغيير، وأن أقصى ما يستطيعه هو تغيير خلفية سطح مكتبه، هذه الآثار لا يعرفها إلا من سبح في فضاء مصادر الحرية، عندما يبحر في عالم يعرف فيه أن كل شيء بين يديه وتحت تصرفه، وأنه لا يفصل بينه وبين تحقيق رغبته وتلبية احتياجاته سوى كتابة أمر أو ضغطة أيقونة أو إجراء تعديل مباشر، سيدرك عندها المعنى الحقيقي للحرية التي تجعل المستخدم يعزز الثقة بنفسه يوماً بعد يوم بعدما كان أسيراً لإجراءات وتراخيص لا يملك إلا الموافقة عليها.



إن النفس البشرية قد جُبلت على حب الحرية والتمرد على القيود، لكن الإشكالية التي تواجهها دائماً هي أن الواقع يخالف ذلك، وبالتالي فإن هذه الحرية سرعان ما تنحسر إبعادها لتتحول إلى حرية نسبية مقيدة تسير وفقاً لما يحيط بها من قيود ومحددات. وبما أن الحديث الذي يجري الآن يتناول المصادر المفتوحة والأنظمة الحرة، فإني أرى من المناسب أن نتناول هذا الجانب، فالمتتبع لتاريخ استخدام أنظمة التشغيل سيجد أننا قد عايشنا فترة من كبت الحريات التي كنا نتوق إليها بل أنه نتيجة لتشبعنا بهذه الأنظمة المقيدة وانعدام البدائل قد تناسينا جانب الحرية الذي كنا نطلبه، والذي تلاشى تقريباً مع موجة الانتشار العارمة لهذه الأنظمة المغلقة فكرياً وتقنياً.

وبما أن لكل فعل ردة فعل معاكسة فقد آن لثورة الحرية التقنية والفكرية أن تثور وتكسر هذه القيود والأصفاد وتنتشر المشاعر التي تضيء دهاليز السجون التقنية والفكرية السابقة.

ففي مجال أنظمة التشغيل المفتوحة وبدءاً من التحكم الكامل بمجريات النظام وعملياته وحتى الوصول إلى البنى الجذرية له، إضافة إلى الانفتاح الفكري والحقوقى لعالم المصادر المفتوحة فإن هذه العوامل قد شكلت معاً اللبنة الأساسية لإيجاد بيئة مستخدمين متفتحة تنمو خبراتهم وعقلياتهم بشكل تراكمي مع تكرار استخدام هذه المنصات العقلية والفكرية والتقنية المتقدمة، وتحولوا من مجرد مستخدمين عاديين إلى عقليات متطورة مطورة.

برنامج Aria2

■ سنتحدث في هذا المقال عن برنامج التحميل aria2 وهو من افضل البرامج اللتي قابلتها باللينوكس وهو يعمل في الواجهة النصيه.

أهم المزايا

- دعم بروتوكولات http https ftp
- دعم الاستكمال بعد انقطاع الاتصال
- تحميل من التورنت
- تحميل من metalink
- تحميل نفس الملف من سيرفرات متعددة
- دعم استخدام الملقم
- دعم الكوكيز

كيفية عمل البرنامج

في حال التحميل من رابط مباشر اكتب من سطر الأوامر :

```
aria2c http://host/file.zip
```

(حيث http://host/file.zip متغير يمثل رابط الموقع)

ملحوظة :

يمكنك أن توقف التحميل مؤقتاً عبر الضغط على Ctrl-C ويمكنك أن تحمل نفس الملف ومن سيرفر آخر لطالما أن الناتج يكون في نفس المكان. فمثلاً إذا أردت التحميل من رابط مباشر وباستخدام اتصاليين نستخدم الصيغة :

```
aria2c http://host/file.zip
```

أما إذا أردت تحميل ملف من أكثر من سيرفر فاكتب :

```
aria2c http://host1/file.zip ftp://host2/file.zip
```

وإذا أردت ان تحمل ملف من سيرفر http وسيرفر آخر ftp فاكتب الأمر :

```
aria2c http://host1/file.zip ftp://host2/file.zip
```

أيضاً يمكنك التحميل من أكثر من رابط موضوعه في ملف نصي txt فاكتب الأمر :

```
aria2c -i files.txt -j 5
```

حيث:

-j هو خيار لتحديد الملفات المطلوب تحميلها وفي حال رغبت التحميل بـ metalink من رابط مباشر فاكتب الأمر :

```
aria2c http://host/file.metalink
```

إذا أردت تحميل metalink من ملف داخل جهازك فاكتب:

```
aria2c -p -t 10 -lowest-speed-limit 4000 -M test.metalink
```

ملحوظة :

يمكنك ان توقف التحميل مؤقتاً عبر الضغط على Ctrl-C ويمكنك المتابعه بنفس الامر ونفس المكان. أما إذا أردت أن تحمل من metalink فقط الملفات المحدده اللتي تريد أن تحملها فاكتب :

```
aria2c -M test.metalink dir/file1.zip dir/file2.zip
```

ملحوظة :

لتعرض الملفات يجب ان تستخدم الخيار -S

كيفية التحميل من ملفات التورنت :

لتحميل ملفات التورنت من ملف تورنت موضوع على سيرفر:

```
aria2c -o test.torrent http://site/file.torrent
```

لتحميل ملفات التورنت من ملف تورنت من داخل الجهاز:

```
aria2c --max-upload-limit 40K -T file.torrent
```

حيث:

```
--max-upload-limit
```

لتعيين السرعة المحدده للرفع.

ويمكنك في أي وقت أن تقوم بإيقاف التحميل بشكل مؤقت من خلال الضغط على Ctrl-C ويمكنك المتابعه بنفس الأمر الذي بدأت به ومن نفس المكان.

لتحميل التورنت وتحديد الملفات اللتي ترغب في تحميلها:

```
aria2c -T test.torrent dir/file1.zip dir/file2.zip
```

يمكنك أن تستعرض الملفات عبر الخيار -S

تحميل التورنت عبر منفذ معين:

```
aria2c -T test.torrent --listen-port 6881
```

لإغلاق البرنامج بعد تحميل التورنت

```
aria2c -T test.torrent --seed-time 120 --seed-ratio 1.0
```

وخيار:

```
--seed-time 120
```

يعني انه سيقفل البرنامج بعد انتهاء التحميل بـ 120 دقيقة.

للتحميل مع تحديد سرعه الرفع:

```
aria2c -T test.torrent --max-upload-limit 100K
```

مصادر الحصول على البرنامج:

يمكن الحصول على آخر نسخة من البرنامج يمكن من خلال الرابط التالي:

```
http://sourceforge.net/projects/aria2
```

أيضاً مستخدمى دبيان والتوزيعات المبنية عليها يمكنهم تركيب البرنامج مباشرة باستخدام الأمر التالي:

```
sudo apt-get install aria2
```

الفيديورا :

```
yum install aria2
```

التركيب من المصدر :

```
tar jxvf aria2-0.*.tar.bz2
```

```
./configure
```

```
make
```

```
su
```

```
make install
```

وللعلم فهناك محاولات لتوفير البرنامج بواجهات رسوميه aria2 من أشهرها :

aria2fe

aria2::gui

karia2

لكن يبقى تشغيل البرنامج الأصلي من سطر الأوامر هو الأقوى، فمثلاً هناك برنامج Mplayer يعمل بواجهة رسومية لكنها لا لاتعطي حتى 10% من إمكانياته الأصلية.

هناك أكثر من أمر يبدأ بنفس الأحرف أو عند الضغط على مفتاح tab لإكمال التلقائي .
الحل هو إضافة السطر التالي على الملف

```
/etc/inputrc :
set bell-style none
```

الآن أعد تشغيل الجهاز وارتح من هذا العناء .

إنشاء ملف xorg في حال عدم تعرف التوزيعية على كرت الشاشة لديك :

حالياً التوزيعات الحديثة تقوم بتعريف كرت الشاشة مباشرة ودون عناء ، ولكن بعض التوزيعات مثل سلاكوير أو بعض التوزيعات القديمة لا تعرف كرت الشاشة مباشرة ، أي لا بد من تدخل يدوي ، استخدم الأمر التالي بالمستخدم root :

X -configure

سيتم إنشاء ملف باسم xorg.conf.new في مجلد home الخاص بالمستخدم root ، تأكد أولاً من أخذ نسخة احتياطية من الملف xorg.conf وتجده على المسار :

/etc/X11

الآن أعد تسمية الملف الجديد xorg.conf.new الى الاسم xorg.conf وارسله الى المسار /etc/X11 وقم بتشغيل الواجهة الرسومية .
قد تحتاج الى بعض التعديلات الطفيفة .

قراءة الملفات العربية في سلاكوير :

لقراءة الملفات العربية في سلاكوير افتح الملف الآتي :

/etc/profile.d/lang.sh

وأضف وأضف UTF-8 سيكون الناتج كالتالي :

export LANG=en_US.UTF-8

البحث في دليل الاستخدام :

إذا كنت تريد البحث عن كلمة معينة ولا تدري في أي manual أي الأمر man تبحث استخدم الطريقة التالية :

man -k mount

سيعرض لك النظام كل manual وردت فيه هذه الكلمة مع رقم الـ manual أيضاً .

التحقق من CD بعد نسخه :

بعد التحقق من أن ملف ISO مطابق من رقم MD5 تقوم بنسخه على CD ولكن مالمذي يضمن لك أن عملية النسخ تمت بشكل سليم . اليك الطريقة :

dd if=/dev/cdrom | head -c 682575872 | md5sum

سيظهر لك رقم مكون من 32 خانة ، قارنة مع رقم MD5 لتوزيعتك المفضله .

دعم السكرول في الماوس على توزيعية سلاكوير

أضف التالي الى ملف xorg.conf

Option "Protocol" "IMPS/2"

Option "Device" "/dev/mouse"

Option "Emulate3Buttons" "yes"

Option "Buttons" "5"

Option "ZAxisMapping" "4 5"

بالطبع تأكد من أخذ نسخة احتياطية من أي ملف قبل تعديله .

تعرف على معلومات جهازك :

لمعرفة معلومات أكثر عن القطع المتصلة بجهازك PCI, USB نفذ الأوامر التالية :

lspci

lsusb

اللينكس يقرأ معلومات الجهاز كاملة وبطريقة صحيحة ولكنه قد لا يعرفها مباشرة ، حتماً ستكون هذه المعلومات مفيدة لك إن كنت تود تعريف القطع يدوياً أو على الأقل ستساعدك في التعرف على عتاد جهازك .
لمعرفة المساحة المتبقية في جهازك والمساحة المستخدمة استخدم الأمر :

df

وهو اختصار لـ Disk Free . ويمكنك إضافة الخيار -h اليه لتكون المعلومات بصيغة ميغا وغيغا ، حرف h مأخوذ من كلمة human أي بشري .

لمعرفة معلومات حول الذاكرة لديك استخدم الأمر :

free -m

سيعرض لك المعلومات بصيغة الميغابايت. وإذا أردت عرض ذلك بالكيلوبايت فاستبدل حرف m بحرف k.
ولمعرفة معلومات عن الأقسام الموجودة على قرصك الصلب استخدم الأمر التالي :

cat /proc/partitions

ولمعرفة المزيد عن المعالج استخدم الأمر :

cat /proc/cpuinfo

ولمعرفة مايجري أثناء اقلاع الجهاز وما تعرضه النواه من معلومات تأكد من الآتي .

- عدم وجود الخيارات التالية في سطر اقلاع
- الكيرنل rhgb أو quiet أو silent أو splash .

مثل هذه الخيارات تحجب عنك الكثير جداً من المعلومات ، ولو كانت هناك رسائل خطأ فلن ترى منها شيئاً . يبدو أنك لاحظت بأن عرض المعلومات كان سريعاً جداً ، كيف أقرأ المحتوى بعناية ؟ بعد اقلاع النظام اقرأ محتوى الملفات التالية وذلك باختلاف التوزيعات .

/var/log/dmesg

/var/log/boot

أو استخدم الأمر :

dmesg

البحث عن الملفات بسرعة :

لجعل عملية البحث عن الملفات أسهل استخدم الأمر :

updatedb

قد يأخذ الأمر وقتاً طويلاً وذلك بحسب عدد الملفات في جهازك ، عند الإنتهاء استخدم الأمر التالي :

slocate filename

locate filename

وذلك بحسب التوزيعية التي تستخدمها .

إيقاف صوت beep في الكونسول :

وهو صوت مزعج جداً وتسمعه عندما تبدأ بكتابة أمر ويكون

لغة البرمجة بايثون Python

لغات البرمجة :

تنقسم لغات البرمجة إلى لغات عالية المستوى high-level language و هناك لغات منخفضة المستوى low-level languages .

اللغات العالية المستوى مثل الـ C/C++ , Java , Python , Ruby .

و سميت باللغة عالية المستوى لأنها الأقرب إلى لغة الإنسان ، فلهذا البرمجة عبارة عن لغة تخاطب بين الحاسب الآلي والإنسان ، و اللغة التي يفهما الحاسب الآلي هي لغة الآلة وهي عبارة عن 010100 المشهورة باسم النظام الثنائي .

ودور المفسرات Compilers هو ترجمة البرامج المكتوبة بلغة البرمجة إلى لغة الآلة وهذا أحد أهم أسباب بطء البرامج المكتوبة باللغات عالية المستوى (في طور البرمجة) مقارنة بنظيراتها منخفضة المستوى مثل لغة الآلة machine language أو لغة التجميع (الأسملي) assembly .

وسيكون حديثنا عن اللغات عالية المستوى .. والتي تنقسم إلى نوعين :

النوع الأول - اللغات التفسيرية مثل Python و Java و Ruby و php والتي تعتمد على مفسر interpreter حيث يقوم المفسر أو المترجم في كل مرة بترجمة البرنامج أمراً أمراً إلى الحاسب الآلي مباشرة :

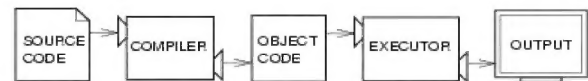


و لهذا يحتاج البرنامج المكتوب بلغة تفسيرية وجود المفسر على النظام . والكود المصدري لهذه اللغة يطلق عليه مسمى script .

النوع الثاني - اللغات التصنيفية ، وهي اللغات تعتمد على مجمع Compiler ومن أمثلة هذه اللغات COBOL , FORTRAN , C ، ودور المجمع compiler يتمثل بقراءة البرنامج ومن ثم ترجمته كاملاً إلى لغة الآلة قبل أن يبدأ البرنامج بالعمل .

و في هذه الحالة يسمى الكود المصدري للبرنامج بإسم source code

و البرنامج الذي تمت ترجمته إلى لغة الآلة يسمى الكائن البرمجي object code أو البرنامج التنفيذي The executable .



وهنا يكون البرنامج المكتوب بلغة هيكلية أو تجميعية أو تصنيفية (بعد التجميع) أسرع من البرنامج المكتوب باللغة التفسيرية ، فيكون كأنه كتب بلغة منخفضة المستوى .

وهناك طريقة جديدة تتبعها بعض اللغات الراقية مثل بايثون و جافا وذلك بنقل الكود المصدري إلى صيغة مكتفة compiled script تختزل بعض العمليات الإجرائية، وفي نفس الوقت لا يتحول النص إلى لغة الآلة بل إلى ملف ثنائي

يسمى byte code وهو عبارة عن سلسلة من تعليمات بلغة الآلة الافتراضية (آلة تخيلية) VM ويكون هناك مفسر يسمى الآلة الافتراضية يتم استدعاؤه في كل مرة يتم تنفيذ البرنامج فيها حيث يقوم بالتحويل من الآلة الافتراضية إلى الحقيقية . ويكون تنفيذ هذا byte code أسرع من النص script ويعتبر تتبع الأخطاء في اللغات التفسيرية أمراً سهلاً لأنها تنفذ سطرًا سطرًا ومن أشهر اللغات التي تتبع هذه الطريقة لغة الجافا .

معنى برنامج :

البرنامج هو عمل سلسلة من الخطوات لحل مشكلة ما .. يعنى ببساطة .. لو أن لدينا مكعب و نريد حساب حجمه .. فنقوم بعمل برنامج يضرب الطول في العرض في الارتفاع .. و يكون هذا حل عام لكل مكعب فيتم كتابة البرنامج فيطلب منك الثلاث متغيرات ويعطيك الناتج لأي مكعب .

مكونات البرنامج :

يتكون أي برنامج من :

- مدخلات .
- عمليات حسابية
- مقارنات منطقية : "أكبر من و يساوي وأصغر من والتكرار"
- مخرجات

لغة البايثون Python :

بايثون لغة برمجة ، من لغات المستوى العالي، تتميز ببساطة كتابتها وقراءتها، سهلة التعلم، تستخدم أسلوب البرمجة الكائنية (OOP) ، مفتوحة المصدر وقابلة للتطوير. تعتبر لغة بايثون لغة تفسيرية، متعددة الأغراض والاستخدامات ، فهي تستخدم بشكل واسع بناء البرامج المستقلة ذات الواجهات الرسومية، وفي عمل برامج الويب، إضافة إلى استخدامها كلغة برمجة نصية للتحكم في أداء بعض البرامج المعروفة او في بناء برامج ملحقة لها. ويمكن أيضاً استخدام البايثون في بناء البرامج البسيطة للمبتدئين، كما يمكن من خلالها إنجاز المشاريع الضخمة كأي لغة برمجية أخرى. غالباً ما يُنصح المبتدئين في ميدان البرمجة بتعلم هذه اللغة لأنها من أسهل اللغات البرمجية تعلماً .

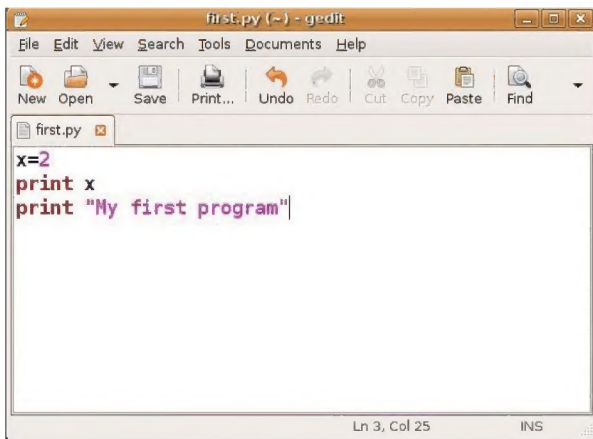
نشأت بايثون في مركز CWI (مركز العلوم والحاسب الآلي) بأمستردام على يد جويدو فان رُزوم في أواخر الثمانينات من القرن المنصرم، وكان أول إعلان عنها في عام 1991. وكتبت نواة البايثون بلغة C. وقد أطلق فان رُزوم اسم "بايثون" على لغته تعبيراً عن إعجابه بفِرقة مسرحية هزلية شهيرة من بريطانيا، كانت تطلق على نفسها اسم مونتى بايثون (بالإنجليزية: Monty Python).

تتميز بايثون بمجتمعها النشط ، كما أن لها الكثير من المكتبات البرمجية ذات الأغراض الخاصة والتي برمجها أشخاص من مجتمع هذه اللغة ، مثلاً مكتبة PyGame التي توفر مجموعه من الوظائف من اجل برمجة الالعاب. ويمكن لبايثون التعامل مع العديد من أنواع قواعد البيانات مثل MySQL وغيره.


```
print x
```

```
print " my first program"
```

ثم قم بحفظ الملف بإمتداد py. مثلا بإسم first.py



ثم أحفظه في أى مكان.. مثلا على سطح المكتب.

ستلاحظ أن بعد حفظ الملف قام البرنامج بتلوين النصوص

هذه الألوان تساعد في كتابة الكود.. يمكنك تظييطها قبل البدء في الكتابة

من قائمة :

View>Highlight Mode>scripts>Python

و أفتح سطر الأوامر terminal و أذهب إلى مكان حفظ الملف و بعد الذهاب إلى سطح المكتب بالأمر cd Desktop

أكتب من سطر الأوامر ومن داخل دليل برنامجك:

```
python first.py
```

```
sherif@ubuntu:~$ cd Desktop
```

```
sherif@ubuntu:~/Desktop$ python first.py
```

```
2
```

```
My first program
```

```
sherif@ubuntu:~/Desktop$
```

ولنا لقاء قادم بمشيئة الله.

لماذا البايثون؟

لغة البايثون متعددة الاستخدامات والأغراض، وقد زاد انتشارها لأنها أكثر لغة جمعت ما بين القوة والسهولة، لذلك نجد أن من أبرز مستخدميها وكالة ناسا NASA :

<http://www.python.org/about/success/usa>

أيضاً من بين كبار مستخدميها موقع مشاركة الفيديو الشهير YouTube.com .. وهذا يدل على أنها لغة يمكن الاعتماد عليها في مجالات بناء وإدارة مواقع الإنترنت.

كما أن من بين مستخدميها المصممين ومستخدمي برامج الجرافيكس.. بالرغم من كونهم ليسوا متخصصين في البرمجة و لكنها اختيرت كأفضل لغة لبرامج التصميم ، ويمكن أن تتكامل مع عدد من برامج التصميم العالمية مثل برامج التصميم ثلاثية الأبعاد مثل مايا Maya و Blender إضافة إلى التكامل مع برامج كثيرة مثل حزمة برامج المكتب OpenOffice.org وبرنامج التصميم الشهير GIMP وهذا فقط للتمثيل وليس الحصر.

برنامجك الأول :

قبل أن نبدأ بالتطبيق نجد أن من المناسب الإشارة إلى موقع البايثون:

www.python.org

وكما ذكرنا فإن اللغة شبيهة بلغة الإنسان الطبيعية، فمثلاً أكتب :

```
print " Hello Linux-ar"
```

شرح هذا البرنامج البسيط ..

كلمة print هي أمر لإظهار الناتج ... فمثلاً قم بوضع x=2 هكذا أنت وضعت قيمة ل x قم بعدها بكتابة print x سيكون الناتج 2 و لكن لو وضعت x بين علامات تنصيص فسيطبعها كنص x هذه هي طريقة إستخدام المفسر مباشرة و لكن هذا للتدريب فقط وهناك الطريقة الثاوية و هي حفظ البرنامج في ملف .. سنستخدم في هذه الطريقة أى محرر نصوص.. هناك فى بيئة الجنوم gedit من alt + f2 و أكتب gedit أو فى بيئة الكيدى kate أو kedit قم بفتح المحرر ثم أكتب فيه :

```
x=2
```

إعداد - علي طوق

الخانة الثالثة : المستخدم صاحب الملف او المجلد

الخانة الرابعة : المجموعة المالكة للملف

الخانة الخامسة : حجم الملف بالبايت

الخانة السادسة : التاريخ و الوقت لإنشاء الملف

الخانة السابعة : أسم الملف

الان سأشرح كيفية فهم خانة الصلاحيات و تقسيمها من خلال الجدول التالي
نقسم عن كل حرف او علامة رقم

1	2	3	4	5	6	7	8	9	10
صلاحيات الآخرين	صلاحيات المجموعة المالكة	صلاحيات المستخدم المالك	نوعية						
-	-	w	x	r	w	x	-	-	-
d	r	w	-	-	-	-	-	w	x
l	r	w	x	r	-	x	r	w	x
d	r	w	x	r	-	x	-	-	-

أظن الجدول واضح و المعاني واضحة

لكنني سأبين مدلولات الحروف في الجدول التالي

صلاحيات الملفات	النوع
قراءة	r
كتابة	w
تنفيذ	x
	ل
	وصلة إلى ملف Link

الان سنتعلم كيفية إعطاء الصلاحيات عبر سطر الأوامر

بطبيعة الحال لدينا 3 مستويات للصلاحيات

المستخدم المالك u

المجموعة المالكة g

الآخرين o

الجميع a

الان نأتي إلى سطر الأوامر

هذا هو الأمر الذي سنستخدمه لتحديد الصلاحيات

chmod

ما صيغة الاستخدام؟

chmod <Permission> <or -> <+> <الخانة>

مكان الخانة نختار هل الصلاحيات للمستخدم أم المجموعة أو

الجميع ... الخ

طبعاً الموجب لإعطاء الصلاحية و السالب لمنعها.

أمثلة على ذلك:

chmod a+x-r hello.text

chmod o+rw hi.c

chmod go-rw private.odt

chmod g+r helloworld.py

```
drwxr-xr-x 17 root root 42280 2007-11-14 03:46
-rwx----- 2 root root 16384 2007-11-14 03:19 log
-rwxr-xr-x 9 root root 4096 2007-11-27 02:37 med
-rwxr-xr-x 2 root root 4096 2007-10-08 13:47 mnt
-rwxr-xr-x 2 root root 4096 2007-10-16 02:17 opt
-rwxr-xr-x 145 root root 0 2007-11-14 05:36 ..
```

Permisson - الصلاحيات

معظم مستخدمي نظامي ويندوز ولينكس لا يعلمون ما هي الصلاحيات، و بالأخص المنتقلون حديثاً الى عالم لينكس لا يعلمون ما فائدة الصلاحيات؟

أنا من وجهة نظري أرى أن الصلاحيات هي جوهر الحماية لبياناتك الخاصة في الحاسوب تحت نظامك،

فكما تعلم أن النظام يتضمن مجموعات و مستخدمين كثر يستطيع المستخدم التحكم فيها بواسطة المستخدم الجذر. فمثلاً إذا لديك مستخدم خاص بك . و هناك آخرون يستعملون نفس الجهاز سواء في المنزل أو العمل أو المدرسة أو الجامعة فبالأكيد لا يرضيك أن يلجوا إلى المعرف الخاص بك، لذلك فبالإمكان أن تعطي كل واحد منهم معرف خاص فيه و بإمكانك انك ان تجعل لكل جماعة من الأشخاص مجموعة لوحدها لها صلاحياتها المحددة وللمجموعة صلاحيات تنقسم لكل معرف لها في مجال صلاحيات المجموعة إذاً موضوع الصلاحيات ينقسم الى قسمين:

- المستخدم
- المجموعة

وسوف نشرح الصلاحيات بشيء من التفصيل:

الصلاحيات تأتي على شكل ثلاثة انواع:

Read صلاحيات القراءة

Write صلاحيات الكتابة

Execute صلاحيات التنفيذ

ملاحظة : كل هذه الاوامر يمكن تنفيذها بصفة المستخدم الجذر.

الآن نذهب إلى سطر الأوامر و ندخل الأمر الذي يقوم باستعراض الملفات على شكل مصفوفة:

ls -l

ستظهر لك النتائج .

الآن قد تتساءل، ما هذه الطلاسم التي ظهرت لي ؟

هذا هو ناتج الامر

```
debian:~# ls -l
total 64
1      2 3 4      5      6      7
drwxr-xr-x 2 ali user 4096 2007-10-27 13:46 a11.zip_FILES
drwxr-xr-x 5 root root 4096 2007-11-19 18:10 Desktop
drwxr-xr-x 8 root root 4096 2007-11-09 15:54 doc
drwxr-xr-x 2 mark user 4096 2007-10-27 19:37 kernel
drwxr-xr-x 4 root root 4096 2007-11-12 21:04 linux_ugfiles_p_files
-rw-r--r-- 1 root root 25397 2007-11-12 21:04 linux_ugfiles.html
-rwxr-xr-x 1 root root 7328 2007-11-12 21:33 speed
-rw-r--r-- 1 root root 224 2007-11-12 21:33 speed.c
-rwxrwxrwx 1 root root 2031 2007-11-14 15:29 youtube-dl
```

سنشرح هذه الطلاسم بالخانات:

الخانة الاولى : خانة الصلاحيات بالأحرف

الخانة الثانية : عدد الملفات و المجلدات داخل المجلد .

اما اذا كانت ملف واحد فقط فيكون الرقم 1

تطبيق الأمر التالي :

```
cat month_random.txt | uniq | sort > mobth.txt
```

الأمر الأول cat لعرض محتويات الملف ، مخرجات هذا الأمر يتم إرسالها عبر بايب لتكون مدخلات على الأمر الثاني uniq والذي يقوم بحذف المدخلات المكرره ، مخرجات هذا الأمر " أي ناتجه " يتم إرساله عبر بايب للأمر sort والذي يعني ترتيب حيث يقوم بفرزها تصاعدياً ، الناتج النهائي يتم إرساله الى الملف month.txt . يمكنك استعراض مكوّناته لترى النتيجة بنفسك .

3 - يدعم wildcards وهي رموز يمكنك استخدامه لتكوين نمط معين ، مثلاً *.txt* تعني كل الملفات التي تنتهي بالإمتداد txt وهذه العلامة * تعني أي سلسلة نصية " لا تحتوي على فراغ " مهما كان طولها . علامة ? وتعني حرف واحد فقط فمثلاً ca? تعني كل الكلمات التي تبدأ بحرفي Ca ثم حرف واحد أو رقم واحد فقط . يمكنك تكوين قائمة بالإحتمالات مثلاً [aeiouy] تعني أي حرف من هذه القائمة . يمكنك تحديد مجال معين [a-z] أي أي حرف من هذا المجال . يمكنك استخدام علامة التعجب للنفى فمثلاً [!abcde] أي لا يحتوي على أي حرف مما في هذا المجال . ويمكنك استخدام كلمات من القائمة فمثلاً :

```
{redhat,suse,debian,slackware}
```

تعني أي كلمة من هذه القائمة ، لاحظ شكل الأقواس وإختلافه عن [] والذي يعني يعني أي ملف يبدأ بحرف من هذه القائمة . وهذه بعض الأمثلة التوضيحية.

```
cp *.rpm rpm/
```

أي نسخ كل ملفات rpm الى مجلد اسمه rpm موجود في الدليل الحالي .

```
rm ca?.txt
```

أي حذف كل الملفات ذات الإمتداد txt التي تبدأ بالحرفين C و a ثم حرف واحد فقط بعدهما.

```
lpr [abcd]*
```

أي طباعة كل الملفات التي تبدأ بأحد الأحرف التالية:

a,e,i,o,u,y .

```
mv [a-m]* /home/mohammed
```

نقل كل الملفات التي تبدأ بأي حرف من هذا المجال من a الى m الى مجلد home الخاص بالمستخدم mohammed . والحال نفسه مع علامة التعجب والتي تعني العكس ، أي ليس مما في القائمة أو المجال .

```
cp {slackware,redhat,debian}*.iso /home/mohammed/iso
```

أي نسخ كل ملفات iso التي تبدأ بكلمة مما في القائمة ونسخها الى مجلد باسم iso موجود في مجلد home الخاص بالمستخدم mohammed .

ملاحظة :

كن حذراً جداً عند التعامل مع wildcards فمسافه واحدة بالخطأ في الأمر التالي ستحذف كل ما هو موجود في الدليل الحالي :

```
rm *.txt
```

هل ترى الفراغ الموجود بين * و .txt. قد تكون تقصد حذف

سطر الأوامر

سطر الأوامر أو Command Line أو التيرمينال أو الطرفية أو الكونسول أو الصدفه كلها مسميات تصب في مصلحة الجزء الأجل والأكثر متعة وفائدة في نظام التشغيل جنو/لينكس . هي ما يميز اللينكس عن غيره من أنظمة التشغيل (ويندوز على سبيل المثال)

فما هو سطر الأوامر؟ وما أنواعه وما مميزاته؟ كل هذه التساؤلات سنتطرق إليها سويًا من خلال هذا الموضوع .

قديمًا لم تكن الأجهزة تحتوي على واجهات رسومية أو ماوس ، ولم تكن هناك أي برامج تعمل من خلال هذه الواجهات ، فكل ما هو موجود مجرد شاشة سوداء و prompt ينتظر منك كتابة الأوامر التي تريدها .

تشغيل البرامج والتفاعل معها والكتابة والنسخ والطباعة تتم من خلال لوحة المفاتيح فقط . بالطبع لم تكن تلك الأجهزة تقوم بنفس المهام التي تقوم بها الأجهزة في الوقت الحاضر، ولكنها كانت تؤدي بعضها، والمجهود فيها مضاعفًا فالخطأ يعني أنك ستبدأ عملك من جديد .

مميزات سطر الأوامر :

- لا يستهلك قدر كبير من الذاكرة .
- يمنحك تحكم كامل في النظام .
- يسهل عليك عملية إدارة النظام عن بعد .
- يمكنك من عمل سكريبتات تقوم بالكثير من المهام دفعة واحدة .
- يمكنك من إصلاح النظام وصيانيته في حال عدم تعرفه على كرت الشاشة لديك أو في حال انهيار الواجهة الرسومية.
- معزول تمامًا عن الواجهة الرسومية وليس كما في أنظمة مايكروسوفت.
- لا يتطلب مواصفات متقدمة ، بل يمكن تشغيله على أي جهاز مهما كانت مواصفاته قديمة.
- في حال كنت تريد إدارة جهاز عن بعد فلا يلزمك سرعة انترنت عاليه لإدارة الخادم.

أدوات سطر الأوامر:

بالطبع ويتم تسميتها صدفية أو shell ولكن أشهرها هو bash . ولأخذ فكرة عامة عن الصدفات الاخرى وتواريخها ومميزاتها ، يمكنك الاستعانة بهذا الرابط :

http://en.wikipedia.org/wiki/Comparison_of_computer_shells

مميزات bash :

- 1 - الإكمال التلقائي ، فليس هناك حاجة لكتابة الأوامر بالكامل وليس هناك حاجة لكتابة أسماء الملفات والمسارات ، يكفي أن تضغط زر tab ليقوم bash بإكمال السطر نيابة عنك أو يعرض لك الإحتمالات في حال توفرها .
- 2 - يدعم خاصية Pip وتعني " الانبوب " ورمزها | وهذه الميزة تمكنك من إرسال مخرجات أمر ما وجعلها مدخلات لأمر آخر ، فمثلاً لديك ملف يحتوي على قائمة بأسماء الأشهر ، مرتبة بشكل عشوائي وفيها بعض المدخلات مكرره، والمطلوب منك هو ترتيبها تصاعدياً وحذف المكرر منها ووضعها في ملف جديد ، على فرض أن الملف الذي يحتوي على القائمة العشوائية هو month_random.txt يمكنك

اختصار ووضع في أحد الملفات التالية :
 /etc/profile أو /etc/bashrc وذلك بحسب التوزيعة
 التي تستخدمها . انظر المثال التالي :

```
alias wget="wget -c -t 0 -T 10"
alias cp="cp -i"
alias rm="rm -i"
```

الآن كل ما احتاجه هو كتابة `wget` ثم الرابط ليقوم النظام باستخدام الخيارات الواردة في الملف . الأمرين الأخيرين هما من باب الاحتياط فعندما يقوم أي مستخدم بمحاولة حذف ملف أو نسخ ملف على ملف آخر بنفس الاسم ، فستظهر له رساله تحذيرية . وهذا حتى لا يحدث خطأ قاتل كما في المثال السابق عندما تحدثنا عن `wildcards` . يمكنك استخدام أي اسم تريد بشرط عدم احتواء على فراغ .
 هذه أبرز مزايا `Bash` . بعضها مشترك بينه وبين أنواع أخرى من الصدقات .
 لماذا يجب علي أن أتعلم استخدام سطر الأوامر ، ألا تكفي الماوس للقيام بمثل هذه المهام ؟

إذا لم تقنعك المزايا السابقة باستخدام سطر الأوامر فإليك هذه الحالات .

- 1 - لديك سيرفر موجود في مبنى مجاور أو في طابق ثاني وحصلت فيه مشكلة ومطلوب منك حل هذه المشكلة ، فكيف ستقوم بذلك ؟ بالطبع من غير سطر الأوامر العملية صعبة جداً .
 - 2 - جهازك الشخصي انهارت الواجهه الرسومية فيه لأي سبب كان ، لديك ملفات مهمة تريد استعادتها ، أو تريد على قراءة ملف `log` لمعرفة سبب المشكلة لعل وعسى أن تجد حل للمشكلة ، أو تريد نقل بياناتك الى جهاز آخر أو قسم آخر أو نسخها على `CD` فكيف ستقوم بذلك دون سابق معرفة بسطر الأوامر . بالطبع هذا في حد ذاته ميزة وهي فصل الواجهه الرومسة عن لب النظام .
 - 3 - لديك جهاز قديم بمواصفات أكل عليها الزمان وشرب ، وأنت تريد تركيب اللينكس عليه لتقوم ببعض الاختبارات (امان ، شبكات ونحوه) بالطبع جهاز بسرعة 100 وذاكرة 64 وقرص 1 غيغا لا يمكنك تثبيت النظام عليه بواجهه رسومية . إذا كيف ستثبت النظام وتديره دون سابق معرفة بسطر الأوامر ؟
- نكتفي بهذا القدر في هذا العدد ، وفي العدد القادم بمشيئة الله سنختار مجموعة من الأوامر لنقوم بشرحها وعرض خياراتها واستخداماتها مع بعض الأمثلة .

كل الملفات ذات الإمتداد `txt` ولكنك بهذا الفراغ تقصد حل كل الملفات الموجودة في هذا المجلد وملف مخفي اسمه `txt` . لاحظ أن الملفات المخفية في عالم لينكس تبدأ بـ `.` عندها سيكون كل شيء قد تم حذفه ثم سيخبرك `bash` بأنه لا يوجد ملف اسمه `txt` .

4 - حفظ آخر الأوامر : هل قمت بكتابة أمر معقد وتريد إعادة تنفيذه ولكنك لا تريد إعادة ما كتبته مجدداً ؟ `Bash` يدعم هذه الخاصية حيث يمكنك استخدام مفاتيح الأسهم (فوق - تحت) لعرض آخر الأوامر التي قمت بكتابتها ، يمكنك أيضاً تنفيذ الأمر `history` لعرض قائمة بالأوامر التي قمت بكتابتها ، يمكنك أيضاً ضغط الأزرار `Ctrl + R` لتغيير شكل `prompt` الى الشكل التالي :
 (reverse-i-search):

الآن اكتب الأحرف الأولى من الأمر ليعرض لك قائمة لاختار منها الأمر الذي تريد .

5 - إعادة التوجيه : رأينا سابقاً عندما تكلمنا عن خاصية `pipe` استخدامنا للعلامة `>` لتوجيه محتويات أمر ما الى ملف معين . هذا هو ما يسمى إعادة التوجيه ، لعلك تسأل مالفرق بينه وبين `pipe` ؟ إعادة التوجيه تقوم بتحويل مخرجات الأمر الى ملف معين أما `pipe` فيقوم بتوجيه مخرجات الأمر وجعلها مدخلات أمر آخر . يوجد أربع أنواع من إعادة التوجيه `<`, `>`, `<>`, الفرق بين العلامة المفردة والمزدوجة هو أن العلامة المفردة تقوم بحذف الملف في حال وجوده وإنشاء من جديد ، أي أن محتويات الملف ستذهب لتحل محلها مخرجات الأمر الجديد . العلامة المزدوجة تنشئ الملف في حال عدم وجوده وتضيف على محتوياته في حال وجوده . أي أنها يشتركان في إنشاء الملف في حال عدم تواجده ويختلفان في التعامل مع الملف الموجود مسبقاً . نأتي الآن الى الفرق بين علامة أكبر من وأصغر .

```
cat /etc/hosts > /etc/hosts.BAK
```

عمل نسخة احتياطية من ملف `hosts` باسم `hosts.BAK` .
`grep 192.168.1.1 < /var/log/httpd/access_log > /root/192.168.1.1`
 استخراج كل الأسطر التي تحتوي على رقم `IP` `192.168.1.1` ووضعها في ملف جديد باسم `192.168.1.1` في مجلد `root` .

6 - دعمه لـ `aliases` : وهي تشبه الاختصارات ، فمثلاً أنا أقوم دائماً بتحميل الملفات عن طريق استخدام برنامج `wget` الذي يعمل من سطر الأوامر ولكنني في كل مره اضطر الى كتابة الصيغة التالية `wget -c -t 0 -T 10` يمكنني عمل

ملاحظة :

الهدف من العملية السابقة هو إعادة تشغيل خدمة NFS وجعلها تعمل تلقائياً عند إعادة تشغيل الجهاز وهذا يختلف بحسب التوزيعية المستخدمة ، ولتزيد من المعلومات راجعوا العدد السابق .

الآن وقد تم تصدير الملف الى الشبكة ، لابد لمن أراد الاستفادة من هذه الملفات بعمل التالي :

```
mount -t nfs server:/path /local-mount-point
```

حيث server هو اسم الجهاز الذي تم تصدير الملفات منه أو رقم الآي بي في حال عدم وجود DNS ، يتبع ذلك نقطتين رأسييتين ثم المسار الذي تم تصديره ، ثم بعد ذلك المسار الذي تود تركيب الملفات عليه ، انظر المثال التالي :
قام الجهاز server-data بتصدير ملفات مشروع ما لجعلها متوفرة لجميع مستخدمي الشبكة ، وقد كانت هذه الملفات الى المسار التالي

```
/media/data/project
```

وتريد أنت تركيب هذه الملفات على المسار التالي في جهازك

```
/media/nfs
```

سيكون الأمر كالتالي :

```
mount -t nfs /server-data:/media/data/project /media/nfs
```

الآن يمكنك تصفح البيانات كما تريد .

هل لدي الصلاحية على تعديل ملفات موجودة على الشبكة ؟

اجابة هذا السؤال تعتمد على أمرين :

1. صلاحيات البيانات المصدرة .
2. صلاحية تركيب هذه البيانات .

لنفترض أن مدير السيرفر قام بتصدير البيانات وجعلها ro أي للقراءة فقط ، فلو قمت أنت بتركيب هذه البيانات بصلاحية ro أو rw فليس لك الصلاحية بتعديل الملفات . أما لو قام مدير السيرفر بتصدير الملفات بصلاحية rw وقمت أنت بتركيبها بصلاحية ro فلا يمكنك تعديلها إلا لو قمت بتركيبها بصلاحية rw .

يمكنك تحديد صلاحية التركيب عن طريق التالي :

```
mount -t nfs -o ro server:/path /local-mount-point
```

اضافة البيانات الى ملف fstab .

يمكنك اضافة بيانات التركيب الى ملف fstab لجعلها تتم تلقائياً عند إقلاع الجهاز .

```
server-data:/path /local-point nfs options 0 0
```

استبدل options بالخيارات التالية :

ro أي للقراءة فقط .

rw للكتابة .

user يمكن للمستخدمين تركيب هذا القسم باستخدام

mount دون وجود صلاحية root .

noauto أي لا يتم تجهيز هذا القسم تلقائياً عند

الإقلاع .



نظام ملفات الشبكة

نظام ملفات الشبكة Network File System أو كما يسمى اختصاراً NFS هو بروتوكول لمشاركة الملفات عبر الشبكة وهو خاص بأنظمة Unix و Linux فقط . تقوم الخدمة على أمرين هما :

- خادم (Server) يقوم بتصدير مجلد معين على الشبكة .
- أجهزة عميلة Clients تقوم بتركيب هذا الدليل .

ملاحظة :

يخطئ الكثير من الناس عندما يعتقدون أن الخادم جهاز خاص بمواصفات خارقة وأسعار فلكية، فالخادم هو الجهاز الذي يقدم خدمة معينة " Web , FTP , Mail , NFS " والأجهزة التي تستفيد من هذه الخدمة تسمى Clients . وهذا يعني أنه قد يكون الجهاز خادم و عميل في نفس الوقت فهو يقدم بعض الخدمات كما أنه يستفيد من خدمات تقدمها أجهزة أخرى .

عندما ترغب في تصدير دليل معين وجعله متاحاً عبر الشبكة فستقوم بالتالي :

- 1 - حدد أولاً الدليل الذي ترغب في تصديره وليكن مثلاً

```
/home/mohammed/project
```

- 2 - بصلاحية المستخدم root عدل الملف التالي

```
/etc/exports
```

وأضف التالي :

```
/home/mohammed/project hostname(options)
```

استبدل hostname باسم أو عنوان الجهاز الذي ترغب في تصدير الملف له ، ويمكنك وضع * لتصدير الملف لكل من في الشبكة المحلية .

الخيارات كثيرة ولكن أهمها هو ro أو rw للقراءة أو للقراءة والكتابة ، SYNC للمزامنة مع الخادم .

الآن نفذ الأمر **service nfs restart** ليقيم النظام

بإعادة تشغيل الخدمة ، ثم نفذ الأمر التالي

```
chkconfig nfs on
```

ليتم تشغيل nfs تلقائياً عند إقلاع النظام .

auto عكس الأمر السابق .

يمكنك automounter من تركيب " عمل mount " لنظام ملفات بمجرد الدخول الى المجلد المسؤول عنه ، وفي حال خروجك من المجلد وعدم استخدامه لمدة معينة تحدها أنت " قد تكون دقيقة " سيقوم النظام بإلغاء تركيب الملفات umount .

الخدمة المسؤولة عن هذه الأداة تسمى autofs . هناك ملفين مسؤولين عن هذه العملية الأول هو auto.master وفيه يتم تحديد مجلد معين " موجود مسبقاً " وتحديد الملف المسؤول عنه وتحديد مدة معينة لإلغاء تركيب الملفات فيه ، الملف موجود تحت الدليل etc .

لنفرض أننا سنحدد الدليل nfs مثلاً ليكون هو الدليل الذي يتم تركيب الملفات فيه ، سنقوم أولاً بإنشاء هذا الدليل تحت / ثم بعد ذلك سنفتح الملف auto.master بأي محرر نصوص ونضيف التالي :

```
/nfs /etc/nfs.auto --timeout=60
```

الحقل الأول هو الدليل الذي سيتم تركيب الملفات داخله ، بعبارة أخرى ستقوم خدمة autofs بمراقبة هذا الدليل وأي عملية وصول له وطلب بيانات منه سيتم فحصها من خلال الملف الذي قمنا بتعيينه في الحقل الثاني " سنقوم بعد قليل بإنشاء وتعديل محتوياته " . ينبغي ملاحظة أن تسميه هذا الملف لا تخضع لنظام معين ، فسمه ماشئت ولكني أفضل أن تكون الطريقة كالتالي اسم الدليل متبوع باللاحقة auto لتعرف أن هذا الملف مسؤول عن مجلد يسمى nfs يتم تركيب الملفات داخله بصورة تلقائية . الحقل الأخير يحدد الوقت الذي يقوم النظام بإلغاء تركيب الملفات بعد انقضاء .

الآن نقوم بإنشاء الملف nfs.auto تحت الدليل etc . قم بإنشاء الملف وأضف داخله للمعلومات التالية :

- 1 - اسم المجلد الذي سيتم إنشاء داخل المجلد nfs وتركيب الملفات فيه .
- 2 - الخيارات التي سيتم تطبيقها عند تركيب الملفات .
- 3 - المسار الذي سيتم تركيب الملفات منه .

انظر المثال التالي لطريقة تعديل الملف :

```
data -rw,soft server:/media/data
music -ro,soft server:/media/music
rpm -ro,soft server:/var/ftp/pub/RedHat
```

السطر الأول هو مجلد سيتم إنشاء تلقائياً تحت الدليل nfs لتركيب محتويات موجودة على الجهاز server في المسار media/data . وهكذا مع باقي الأسطر .

ملاحظة مهمة جداً :

لاتقم بإنشاء المجلدات data و music و rpm أبداً .

الآن قم بإعادة تشغيل الخدمة autofs بالأمر التالي :

```
service autofs restart
```

ثم قم بجعلها تعمل تلقائياً عند بدء التشغيل بالأمر التالي :
chkconfig autofs on

كيف أعرف الملفات المصدره من جهاز بعيد ؟
في الحالات السابقة كنا نعلم المسار الذي قام مدير السيرفر بتصديره . ماذا لو لم تكن نعلم ذلك أو كنا في حالة شك . الحل هو باستخدام الأمر

showmount

طريقة استخدام الأمر السابق هي كالتالي :

```
showmount -e server
```

كما ذكرت سابقاً يمكنك استبدال server برقم الآي بي في حال عدم وجود DNS .

أنواع تركيب ملفات NFS .

هناك ثلاث أنواع لتركيب ملفات NFS وهي

• hard

• soft

• interruptible

النوع hard وهو الافتراضي .

عندما تقوم بتركيب ملفات NFS على الوضع الافتراضي hard ، والذي سيحدث في حال توقف السيرفر لأي سبب كان ، أو توقف الشبكة لأي طارئ ؟ في هذه الحالة سيتوقف جهازك عن العمل على هذه الملفات ولن تستطيع إيقاف العمليات التي لها علاقة بهذه الملفات حتى مع استخدام ctrl + C أو استخدام الأمر kill وسيظل الكيرنل يحاول الوصول الى الملفات حتى يستطيع أو ستظطر الى إعادة التشغيل .

النوع interruptible .

نفس الحالة السابقة ولكن في هذه الحالة يمكنك استخدام ctrl + c و kill لإيقاف العملية .

النوع soft .

في هذه الحالة سيقوم الكيرنل بمحاولة الوصول الى الملفات لمدة 30 ثانية ثم سيعرض لك رسالة خطأ .

كيف أحدد نوع تركيب ملفات NFS ؟

يمكنك تحديد نوع التركيب عن طريق الخيار O انظر المثال :

```
mount -t nfs -o intr.....
```

```
mount -t nfs -o soft .....
```

لماذا يكون الوضع hard هو الافتراضي ؟

لأنه هو الوضع الوحيد الذي تضمن فيه كتابة البيانات على الجهاز البعيد ، لنفترض أن موظفاً في البنك قام بعملية إيداع وكانت كل البيانات تخزن على جهاز بعيد ! أعتقد أنك عرفت السبب .

حسناً هل هناك حل لهذه المشكلة ؟

بالطبع والحل هو استخدام Automounter .

2 - ينتقل النظام الى الملف السابق ويسأل عن الملف المسؤول عن هذا الدليل " في هذه الحالة الملف موجود تحت الدليل etc بإسم nfs.auto .

3 - ينتقل النظام الى الملف السابق ليسأله هل هناك أي شيء يتعلق بالمجلد project . في حال كانت الإجابة بنعم سيقوم بإنشاء هذا المجلد بحسب الخيارات الموجودة في الملف ، وفي حال كانت الإجابة بلا ستعرض لك رسالة خطأ .

بإمكانك التعامل مع الأقسام المحلية لقرصك الصلب ولكن مع بعض الفروقات البسيطة ، انظر المثال التالي :

```
mydata -fstype=ext3 :/dev/hda5
```

ولكني أعتقد أن استخدام automounter سيكون أفضل مع الملفات المشتركة على الشبكة ، فلا حاجة لإستخدامة مع أقراص محليه موجودة على نفس الجهاز .

الآن من سطر الأوامر حاول استعراض محتويات الدليل nfs ماذا ترى ؟
بالتأكيد الدليل فارغ ولكن نفذ التالي :

```
ls /nfs/data  
ls /nfs/music  
ls /nfs/rpm  
ls /nfs  
ls /nfs/projec
```

ستجد أنه مع الثلاث الأوامر الأولى تم عرض محتوياتها ، فقط قام النظام تركيبها يمكنك تنفيذ الأمر mount لترى أنه قد قام بتركيبها فعلاً . الأمر الرابع سيعرض لك ثلاث مجلدات تحت الدليل nfs ولكن إن انتظرت لمدة دقيقة ثم حاولت استعراض محتويات الدليل nfs فلن تجد شيئاً . الأمر الأخير سيعرض لك رسالة خطأ ، لنحاول أن نشرح ماالذي حدث :

1 - هناك عملية process تقوم بمراقبة الدليل nfs وأي عملية تتعلق بهذا الدليل سيتم السؤال عنها في الملف . auto.master

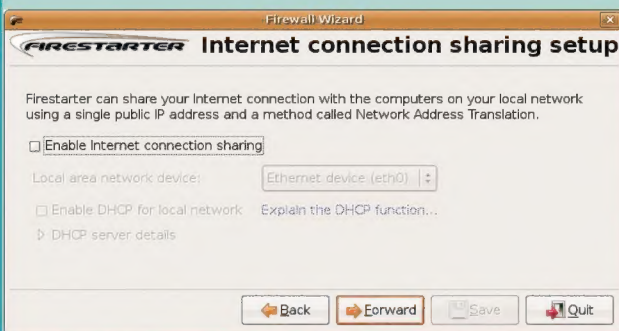
إعداد - أبو محمد العسيري

● نضغط التالي (forward) لتظهر الشاشة التالية: الشكل (1-2)



الشكل (1-2)

● نختار من القائمة الكرت المراد العمل عليه (المشبوك بالنـت) والخيار الثاني لمراقبة الاتصالات الخارجة من الجهاز ثم نضغط التالي (forward) لتظهر الشاشة التالية: الشكل (1-3)



الشكل (1-3)

■ إذا كان جهازك يوزع النت تختار الكرت الذي عليه المشاركة وإلا فاترك الخيار بدون تحديد ونضغط التالي لتظهر لنا آخر شاشة كالتالي: الشكل (1-4)

رغم أن المستخدم المنزلي لا يحتاج إلى جدار النار في لينكس غالباً ، إلا أن كثيراً من المستخدمين الجدد يسأل عن الفايروول ، هذا شرح مبسط لاستخدام جدار النار Firestarter وهو واجهة رسومية للفايروول iptables الذي تعرض له الأخ محمد عبدالله في مقالة سابقة. يكون تثبيت البرنامج بالنسبة لمستخدمي أوبنتو وغيرها من التوزيعات القائمة على apt-get كالتالي:

```
sudo apt-get install firestarter
```

وغيرهم بإمكانه تنزيلها من مواقع الحزم الخاصة بتوزيعه أو من الأقراص أو من الملف المصدري في الموقع <http://sourceforge.net/projects/firestarter/>

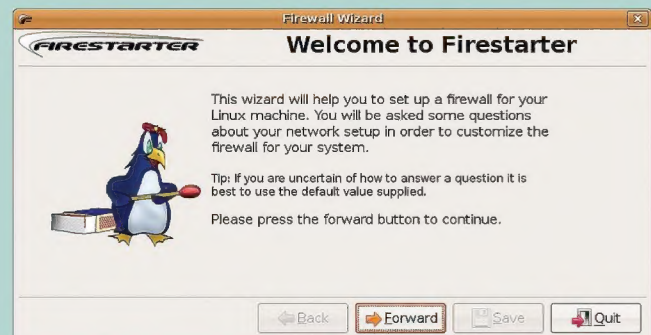
● بعد تثبيته تشغله من الواجهة

firestarter > Administration > System

أو من سطر الأوامر

```
sudo firestarter
```

● ليظهر معالج الإعداد (عند التشغيل للمرة الأولى) كما في الشكل (1-1)



الشكل (1-1)

إعداد - أبو محمد العسيري

حدثت منع اتصال أو تحذير منه. أما الثالث فلعرض قوانين الفايروول والتعديل عليها. بطبيعة الحال عندما تختار من القائمة (في لسان التبويب Policy) البكتات الصادرة فتجد القاعدة هي السماح ويمكنك تعديلها إلى منع الجميع والسماح لاتصالات محددة (أمان عال) وتضيف ما تريد السماح له في القوائم السفلى بضغط الزر الايمن واختيار Add Rule أو تركها كما هي وهو الافتراضي. أما لو اخترت البكتات والاتصالات الواردة فالقاعدة هي المنع وإذا أردت السماح لأحد تضيفه في القائمة السفلى.

■ لإعادة إظهار معالج الإعداد نختار من القائمة العلوية في البرنامج Firewall

ثم
RUN Wizard

■ لجعل البرنامج يعمل مع بدء التشغيل بالنسبة لأوبنتو نذهب إلى

System ----> preferences -----> sessions

ثم نضيف عنصر جديد ونسميه
firestarter

وتضع الأمر
gksudo firestarter

في المربع
command

وللتوزيعات الأخرى من سطر الاوامر وانت root:

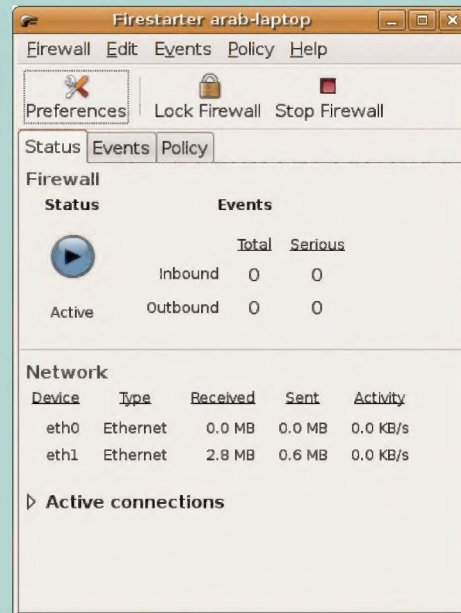
```
lnln -s /etc/init.d/firestarter /etc/rc5.d/S99firestarter
```

طبعا مع تغيير رقم المجلد 1C لرقم مستوى التشغيل في نظامك.



الشكل (1-4)

● نضع صح على الخيار الموجود ثم نختار save لحفظ الإعدادات وبدء الفيرول وتظهر الشاشة الرئيسية كالتالي: الشكل (1-5)



الشكل (1-5)

■ ترى في الأعلى ثلاث إيقونات:

الأولى من اليسار لتعديل اعدادات البرنامج

والثانية لغلاق ومنع جميع الاتصالات بينما

الثالثة لإيقاف وتشغيل الفايروول. كما يظهر

أسفل الأيقونات ثلاثة ألسنة تبويب: الأول لعرض

حالة الاتصالات الحالية والثاني لعرض الاحداث التي

Pingus



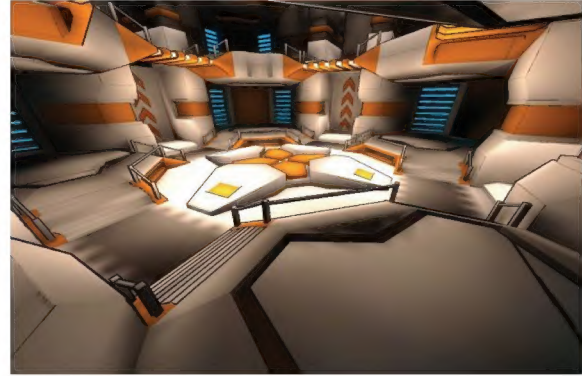
الترخصة : GPL

الأنظمة : جنو/لينكس ، مايكروسوفت ويندوز
موقع اللعبة : <http://pingus.seul.org>

■ لعبه حرة اخرى، مشابهه للعبة الليمنجس Lemmings-like الموجودة في أنظمه ويندوز. تحتوي اللعبة على 22 مستوى. بداية اللعبة كانت في سنة 1998 وهدفها انتاج لعبة مشابهة للعبة الليمنجس Lemmings-like، من ثم بدأت تنتشر اللعبة ولأنها مفتوحة المصدر كان العديد من المتطوعين يشاركون فيها حتى أصبح هناك فريق عمل لكل شيء في اللعبة، تحتوي لعبة بينجوس على العديد من الأفكار والإبداع التي لا تتوافر في ليمنجس.



Warsow



الفئة : اكشن (ار بي جي)

الترخصة : GPL

الأنظمة : جنو/ لينكس ، مايكروسوفت ويندوز
موقع اللعبة : <http://www.warsow.net>

■ تعتمد لعبة وارسو على القتال الفردي 3D FPS ، وجمعت ما بين الإثارة والجودة العالية. تعمل اللعبة تحت محرك Qfusion وهو عبارة عن محرك مطور لمحرك Quack2. رسوم اللعبة شبيهة برسوم الكرتون والمازجا وتمزج بينها وبين مؤثرات متعددة لتعطي رونقاً وطعماً للعبة.

Sauerbraten



الفئة : اكشن

الترخصة : ZLIB

الأنظمة : جنو/ لينكس

موقع اللعبة : <http://sauerbraten.org>

■ تصنف هذه اللعبة مع فئة آر بي جي 3D FPS ، وتعتمد في أسلوبها اللعب الفردي والمشارك . بنيت اللعبة على محرك Cube القوي. هدف اللعبة ليس فقط اللعب وانتظار تطويرها من الفريق، بل الهدف الاساسي هو ان تعطي الحرية للمستخدم في التدخل بالتعديلات وبناء اللعبة حسب احتياجاته والتعلم منها وتطويرها.